NUI Galway
OÉ Gaillimh

Autumn Examinations 2016 / 2017

Exam Code(s)	4BCT1, 4BP1
Exam(s)	B.Sc. Degree (Computer Science & Information Technology) Bachelor of Engineering (Electronic and Computer Engineering)
Module Code(s)	CT417
Module(s)	Software Engineering III
Paper No.	1
External Examiner(s)	Dr. John Power
Internal Examiner(s)	Dr. Michael Schukat *Dr. Enda Barrett *Dr. Sam Redfern

Instructions: Answer question 1 (mandatory) and any 3 other questions.
Answer 4 questions in total.
All questions carry equal marks.

Duration	2hrs
No. of Pages	6 (Including cover page)
Department(s)	Information Technology
Requirements	None

Question 1 is mandatory

1: Consider a web-based application: a social media site. The application must track users, their friends, posts and 'likes'. (A 'like' is something that a user applies to someone else's post).

(a) Describe at a high level (and with the aid of a diagram) a layered (n-Tier) architecture design which would meet the basic requirements of the system. Include a brief list of typical responsibilities for each layer.

8 Marks

(b) Describe the logic and information flows through the system according to these two use cases:

- (i) A previously registered user logs in and is presented with the site's home page.
- (ii) A user searches for another user by name, and is presented with a page of results.

4 Marks

(c) Assume that the social media site becomes incredibly successful and has to deal with an increasing number of transactions per second.

- (i) Where are the bottlenecks in the system likely to occur? – i.e. which parts of the system will be unable to cope with the demands on them.
- (ii) Describe an alternative software architecture which could be used to alleviate these bottlenecks (there are several to choose from)
- (iii) Briefly, what new challenges would this architecture bring?

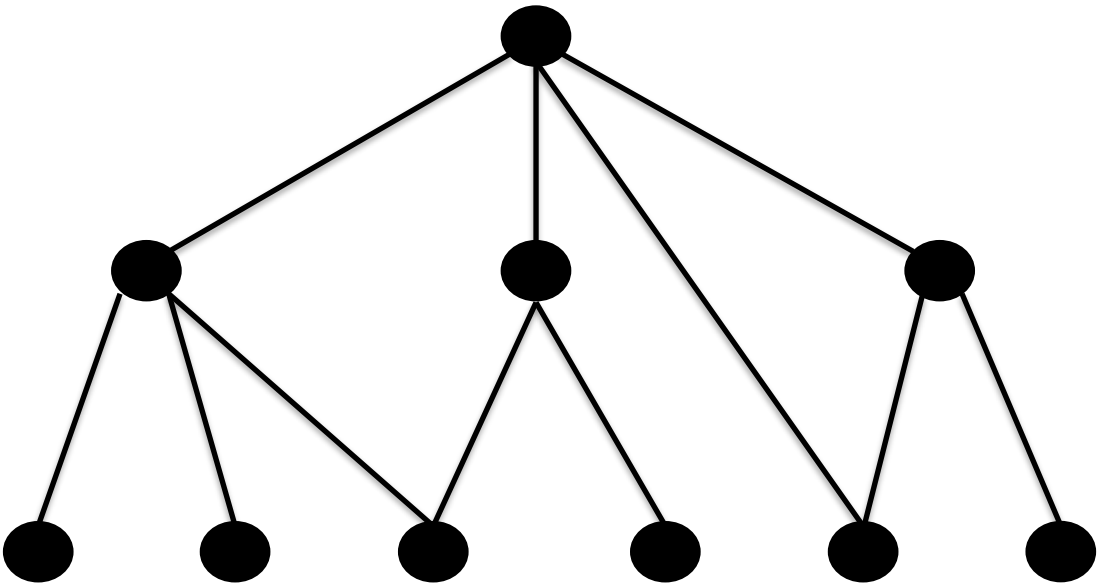
8 Marks

2: (a) Identify the spanning tree for the following software module design, and calculate the values for Tree Impurity (m(G)) and Internal Reuse (r(G))

Remember:

$$m(G) = \frac{number\ of\ edges\ more\ than\ the\ spanning\ tree}{maximal\ number\ of\ edges\ more\ than\ the\ spanning\ tree}$$

$$r(G) = number\ of\ edges\ additional\ to\ the\ spanning\ subtree$$



12 Marks

(b) Explain why tree impurity m(G) is a useful measure for assessing the potential quality of a software architecture.

4 Marks

(c) Give two limitations of using the internal reuse metric r(G) for software module design

4 marks

PTO

3: (a) Define the main assumption underlying the Jelinski-Moranda (JM) model of software reliability. Clearly show the formulation for the hazard rate. Why is the JM model suitable as a model of software reliability growth?

Assuming the initial number of faults (N) in the system is 7, predict the MTTF for the system after each successive system repair. Assume that $\varphi = 0.005$, where φ is the contribution of each fault to the failure rate.

Plot the sequence of MTTF values and comment on the shape of the curve.

8 Marks

(b) A software component fails on average once every 15 days. Assuming a probability density function based on the exponential distribution calculate:

- The hazard rate of the system
- The probability that the system will fail in the first 5 days of operation
- The reliability of the system after 25 days of operation

12 Marks

4: (a) Define the representational condition for the following empirical relations between software system A and software system B. Clearly show the relational mapping between the real world and the number system. For each scenario, select an appropriate number system value that can be used.

- A is more reliable than B
- A is larger than B
- A is more costly than B
- A requires more computational resources than B

10 Marks

(b) Describe, using examples, the following object-oriented measures:

- Coupling between objects
- Weighted methods per class

6 Marks

(c) In measurement theory, discuss the difference between *direct* and *indirect* measurements.

4 Marks

PTO

5: (a) For the following class, calculate the Lack of Cohesion of Methods (LCOM) measure.

```
class Account
{
      String id;
      double balance;
      double RATE = 1.11;

      public getID(){ return this.id; }

      public getBalance(){ return this.balance; }

      public credit(double amt)
      {
            this.balance+=amt;
      }

      public debit(double amt)
      {
            this.balance-=amt;
      }

      public getExchangeRate(){ return this.RATE; }

      public setExchangeRate(double v)
      {
            this.RATE=v;
      }
}
```

16 marks

(b) Discuss the result, and comment on the strengths and weaknesses of the LCOM approach.

4 Marks

PTO

6: (a) Describe the software development practice of continuous integration, include in your answer the benefits of building software this way.

4 Marks

(b) Discuss the difference between branch and line coverage. In the following code example calculate the branch and line coverage produced by a test where isLoggedIn is set to true

```
public void loginStatus(boolean isLoggedIn)
  {
    if(isLoggedIn)
    {
      System.out.println("User is logged in");
    }
    else
    {
      System.out.println("User is not logged in");
    }
  }
```

10 Marks

(c) Explain the purpose of the Project Object Model (POM) for a Maven project. Maven supports inter-project relationships through dependencies. Can you explain what a project dependency is and also what transitive dependencies are?

6 marks

End